CHAPTER

6

WINDOWS 2000 SECURITY AND ACCESS CONTROLS

After reading this chapter and completing the exercises, you will be able to:

- Describe the Windows 2000 security model, and the key role of logon authentication
- Customize the logon process
- Discuss domain security concepts
- Understand the local computer policy
- Enable and use auditing
- Encrypt NTFS files, folders, or drives, using the encrypting file system (EFS)

Because Windows 2000 plays a pivotal role on so many networks, the operating system has been constructed to provide a wide range of control over access to its resources. In fact, Windows 2000 Server has been designed to be able to check access permissions for every request before granting access to resources. This chapter will explore the details of the Windows 2000 security model, its logon process, and the ways in which the operating system associates security information with all objects under its control. You will also see that any user's or program's request for system resources is subjected to close scrutiny at blinding speeds.

THE WINDOWS 2000 SECURITY MODEL

Windows 2000 Professional can establish local security when used as a standalone system, or participate in domain security (managed by Windows 2000 Server, Windows NT Server, or some other NOS). Before a user can use any Windows 2000 resource, he or she must log on to a system, a workgroup, or a domain by supplying a valid user ID and **password**. A user who successfully logs on receives an **access token**. The access token includes information about the user's identity, any permissions specifically associated with the user's account name, and a complete list of all the groups to which the user belongs—including custom groups defined for your network and Windows 2000 predefined default groups. A complex string of bits represents the token, which is attached to every **process** that the user initializes until that user logs off. In other words, each time a user runs a program, enters a system command, or accesses a resource, a copy of that user's access token accompanies the request.

Each time a user attempts to access a resource, the user's access token is compared with a list of permissions associated with the resource. This list is called an **access control list (ACL)**. The access control list is one of the more important attributes associated with any Windows 2000 resource. Whenever an object is requested, the ACL and the access token are carefully compared, and a request for the object is granted only when a match is found.

If the system finds a match between the access token and the ACL, the request can proceed. If a Deny permission occurs or if the requested service is not permitted, the request is denied. A match between the access token and the ACL is like a key that fits a particular lock. In fact, many experts explain the access token using the analogy of a ring of keys that you try in a lock one at a time until a match is found or until there are no more keys to try. Matches between an access token and the ACL can be a function of permissions associated with the individual user's account or permissions that derive from the user's membership in a local or global group. Whatever the source of the permissions, the user's request is allowed to proceed unhindered if a match is found.

Windows 2000 domain security is centered around **Active Directory**, the centralized database of security, configuration, and communication information maintained by domain controllers in a Windows 2000 network. Active Directory supports everything from authentication of domain users' accounts to accessing shared resources. Windows 2000 Professional as a standalone desktop system does not use Active Directory, relying instead on the Registry and internal security systems (similar to Windows NT 4.0) to control user access. However, Windows 2000 Professional participates in Active Directory when it is used as a client in a Windows 2000 domain network. All of the information about the domain and all of the resources shared by the network are managed by Active Directory. Windows 2000 Professional uses Active Directory to gain access to the domain network. Both domain and local security use logon authentication, objects, and access control to gain access to Windows 2000 resources.

TIP Because Active Directory is not a Windows 2000 Professional feature or component, it is not discussed in great detail here. However, an overview of Active Directory is included in Appendix B, "Working with Active Directory."

Logon Authentication

Windows 2000 logon is mandatory to gain access to the system and to applications and resources. The logon process has two components: identification and authentication. **Identification** requires that a user supply a valid account name (and in a domain environment, the name of the domain to which that **user account** belongs). **Authentication** means that a user must use some method to verify his or her identity. By default in Windows 2000, possession of the proper password for an account constitutes authentication, although Windows 2000 also supports third-party authentication add-ins, including biometric systems that check fingerprints or perform retinal scans, and smart card systems that require physical possession of a unique electronic keycard to prove a user's identity.

TIP Most typical Windows 2000 systems rely solely on passwords for authentication, so using hard-to-guess passwords is an important aspect of good system security. Good passwords generally include both uppercase and lowercase letters as well as numbers, for example, Ag00dPA55w0Rd. By creating passwords such as this, it becomes impossible for programs that attempt system break-ins to gain access by using dictionary lists to search for valid passwords.

When a user successfully logs on to a Windows 2000 machine, the security subsystem working with Executive layer services creates an access token for that user. The access token includes all security information pertaining to that user, including the user's **security ID (SID)** and SIDs for each of the groups to which the user belongs. Indirectly, through the user rights policy, this collection of SIDs informs the system of the user's rights. An access token includes the following components:

- The unique SID for the account
- A list of groups to which the user belongs
- A list of rights and privileges associated with the user's specific account

Access to the system is allowed only after the user receives the access token. Each access token is created for one-time use during the logon process. Once constructed, the access token is attached to the user's **shell** process, which defines the run-time environment inside which the user executes programs or spawns other processes. (The default shell process for Windows 2000 is Windows Explorer. It defines the desktop, Start menu, taskbar, and other elements of the default user interface. Alternate shells from third parties can be employed, or even the Windows NT 3.51 Program Manager can be used.) As far as Windows 2000 is concerned, a process is a computer program designed for some specific function. A process is a term synonymous with program. Every activity within the user mode and kernel mode is performed by a process. Each process is launched using the access token of its parent—that is, the process that caused it to be launched. When a user launches a process manually, he or she does so through the shell process, usually Windows Explorer, and it is the access token of the shell process that is inherited by the new process.

Objects

In Windows 2000, access to individual resources is controlled at the **object** level. Each object hosts its own access control list (ACL), which defines which users and groups have access permissions and exactly what type of access they are granted (read, write, print, modify, list, etc.). Everything within the Windows 2000 environment is an object; this includes files, folders, processes, user accounts, printers, and computers. Requests for resources, therefore, translate into requests for objects. An individual object is identified by its type, which defines its permitted range of contents and the kinds of operations (called services) that may be performed upon it. Any individual object is an instance of its type and consists of data and a list of services that can be used to create, manipulate, control, and share the data it contains.

Windows 2000 is able not only to control access at the object level, but also to control which services defined for the object's type a particular security token is allowed to perform or request. All objects are logically subdivided into three parts: a type identifier, a list of services or functions, and a list of named attributes, which may or may not have associated data items—called values.

When defining an object, its type describes the kind of entity it is. For example, an object's type may be file, directory, printer, or network share. An object's services define how the object can be manipulated; for example, possible services for a directory object are Read, Write, and Delete. An object's attributes are its named characteristics, such as *filename*, read-only, hidden, file size, and date created for an object whose type is file. The values for these attributes are their content, such as the actual name of the file, selected, not selected, 142,302 bytes, and 10/3/99 04:23:34 PM, respectively.

Remember, access or permission to use to an object is determined on the basis of the entire object and also for each of the services defined for that object. For example, a user can have access to read a file, such as an e-mail program executable file, but not to edit or delete it. Thus, users can have permission to access the object in general, but may have more specific controls about which services they can request in connection with that access.

CAUTION

Keep in mind that Windows 2000 automatically grants the Everyone group Full Control to the object whenever a new object or share is created. Thus, you must implement restrictions on new objects and new shares. In other words, Windows 2000 allows everyone access to new objects by default.

Access Control

The Windows 2000 logon process is initiated through the attention sequence (the Ctrl+Alt+Delete keystroke combination, known to many DOS users as the "three-fingered salute"). This attention sequence initiates a hardware interrupt that cannot be "faked" by a program and brings up a logon procedure dialog box that is stored in a protected area of memory, thus securing the system from attack through an unauthorized remote logon.

This combination of characteristics for the logon authentication procedure is the key to the entire Windows 2000 security scheme, because all other security features are based upon the level of authority granted a user who has successfully logged on. The Windows 2000 security structure requires a user to log on to a computer with a valid username and password. Without this step, nothing more can be accomplished in the Windows 2000 environment.

The Windows 2000 logon procedure provides security through the use of the following:

- *Mandatory logon:* The user must log on to access the computer.
- *Restricted user mode:* Until a successful logon takes place, all user-mode privileges are suspended. Among other things, this means that the user cannot launch applications, access resources, or perform any action or operation on the system.
- *Physical logon:* The structure of the logon sequence ensures that the logon occurs from the local keyboard, rather than from some other internal or external source. This is because the attention sequence initiates a hardware interrupt that accepts input only from the local keyboard.
- *User profiles:* Windows 2000 allows each user who logs on to a particular machine to save user preferences and environment settings, called a **user profile**. Each user can have a set of specific preferences restored at logon or can be supplied with a mandatory or default set, depending on how the system is configured. A user profile that is configured to follow a user throughout a network is called a roaming profile. (Profiles are covered in detail in Chapter 5.)

Customizing the Logon Process

A system administrator can alter the default logon process appearance and function using WinLogon. The **WinLogon** process produces the logon dialog box in which the username, password, and domain are selected. WinLogon also controls automated logon, warning text, the display of the Shutdown button, and the display of the last user to log on to the system. WinLogon operates in the user mode portion of the Windows 2000 system architecture and communicates with the Security Reference Monitor and SAM database in the kernel's Executive Services to authenticate users and launch their environment shell with an attached user-specific access token. The WinLogon process can be customized to display some or all of the following characteristics:

- Retain or disable the last logon name entered
- Add a logon security warning

- Change the default shell
- Enable/Disable the WinLogon Shutdown button
- Enable automated logon

> TIP
>
> Most of these characteristics can be altered through the Local Security Policy, accessed from the Control Panel, Administrative Tools, Local Security Policy (see Chapter 5). All of these characteristics can be controlled through the Registry (see Chapter 13) via the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\WinLogon key. However, Microsoft recommends using the Local Security Policy interface to alter these items instead of the Registry when possible.

Several possible configuration changes for the Windows 2000 logon process are detailed in the following sections.

Disabling the Default Username

By default, the logon window displays the name of the last user to log on. If the same user consistently logs on to a single machine, displaying the logon name is convenient; however, for shared or public-access machines, this provides a key piece of information that someone could use to break into your system. It is possible to change the default by altering the value of its associated Registry key (DontDisplayLastUserName) or by setting a Local Security Policy value. Another common form of attack is a dictionary attack, which involves supplying the contents of a dictionary, one word at a time, as a logon password. Avoiding such systematic break-in attempts explains why it's a good idea to limit the number of failed logon attempts via the Lockout Policy.

Disabling the default username option presents a blank username field at the logon prompt. Note that the related value and its corresponding assignment do not occur in the Registry by default. The value is named DontDisplayLastUserName, and it is of type "String," where a value assignment of 1 disables the name display and a value of 0 enables it. This control appears by default in the Local Security Policy (as noted, it is recommended that you use the Local Security Policy too to manage this feature rather than edit the Registry). (Try Hands-on Project 6-2 to disable the display of the username of the last successful logon.)

Adding a Security Warning Message

Depending on your organization's security policy, you might be legally obligated to add a warning message that appears before the logon prompt is displayed. U.S. law states that if you want to be able to prosecute individuals for unauthorized entry to or use of a system, you must warn all users that usage is monitored, unauthorized access is forbidden, and that unauthorized users might be liable for prosecution.

Two Registry or Local Security Policy values are involved in this effort:

- *LegalNoticeCaption*: Puts a label on title bar of the legal notice window that appears during logon. This field works best with 30 characters or less of text.

- *LegalNoticeText*: Contains text information that provides the details of the warning to be issued to system users. This field may be up to 65,535 characters long, but most warning messages do not exceed 1000 characters in length.

After this feature has been activated and configured, a warning message appears each time a user enters the Windows 2000 attention sequence. This message requires the user's acknowledgment by pressing OK before the logon window displayed. (Try Hands-on Project 6-3 to add a legal notice to your logon.)

Changing the Shell

The default shell (the application launched by WinLogon after a successful logon) is Windows Explorer. You can change the shell to a custom or third-party application, depending on the needs or security policy of your organization. For example, you could change to the use of the Program Manager familiar to NT 3.51 and Windows 3.x users. To make this change, you change the Shell value in the WinLogon key from Explorer.exe to Progman.exe.

CAUTION

> If you change from the Explorer shell to the Program Manager shell, your system will lose its onscreen taskbar, and you will no longer be able to use the Start menu to launch programs from your desktop. For this reason, most organizations use the default shell.

If you use the default shell, remember that one of the options that appears on the WinLogon window is Task Manager. This tool is available any time you enter the attention sequence. From there, you can select the Applications tab, and New Task (Run) from the File menu lets you browse and launch applications from the desktop. (Try Hands-on Project 6-4 to change your default shell.)

Disabling the Shutdown button

By default, the Windows 2000 Professional logon window includes an enabled Shutdown button. However, in an environment in which users have access to the keyboard and mouse on a Windows 2000 machine, this option has the potential for unwanted system shutdowns. Fortunately, this option can be disabled. It should be noted, however, that if the user still has access to the physical power switch on the computer, disabling this option might cause more headaches than it solves. A system that has been shut down or rebooted through the operating system has a much higher chance of coming back up successfully than one that was simply powered off. Note that by default, this button is enabled for Windows 2000 Professional machines, but disabled for Windows 2000 Server machines.

The value named ShutdownWithoutLogon is the one you'll need to edit in either the Registry or the Local Security Policy. It's enabled (set to the value 1) on Windows 2000 Professional machines by default. To disable this button, change its value assignment to 0 (zero); to reenable it, reset its value to 1. When the button is disabled, it still appears in the WinLogon window, but it's grayed-out and unusable.

For laptops or other advanced computers with automatic shutdown capabilities, an additional button labeled "Shutdown and Power Off" appears. Similar machines might also support a sleep mode, in which all processing is suspended and all power turned off, except to the computer's RAM. In that case, Sleep also shows up as a shutdown option. This particular setting permits users to eliminate most of a computer's power consumption, yet be ready to resume activity at the push of a single button or movement of the mouse. If the machine warrants such settings, users will find related Registry values in their WinLogon key settings that help them control how these functions are handled.

CAUTION

Be aware that leaving the Shut Down button enabled means that anyone with access to the keyboard can enter the Windows attention sequence and shut down the local machine.

Automating Logons

Some special- or limited-use Windows 2000 machines (for example, airport kiosks or hotel information stations) may need to be always available and always logged into a low-security account for access to some dedicated application. Although the logon process cannot be bypassed, the values for username and password can be coded into the Registry to automate logons. This normally is of interest only when installing machines for public use, such as for information centers, kiosks, museum guides, or other situations in which a computer is used to provide information to the public. In such cases, it's important to have computer and user policies to prevent Windows-2000-savvy users from attempting to break out of the public application and explore other, less-public aspects of the system (or worse, of the network to which it may be attached).

To set up an automated logon, the following Registry keys must be defined and set:

- *DefaultDomainName:* Defines the name of the domain to log on to (needed only when logging on to a networked machine that's part of a domain)
- *DefaultUserName:* Defines the default logon account name
- *DefaultPassword:* Defines the password associated with the default account name. This value is not present by default. When auto logon is disabled, delete this value, because it stores the password in plain text.
- *AutoAdminLogon:* Instructs the machine to log itself on immediately following each bootup. A value of 1 automatically logs on, using the credentials from the other three values in this list. A value of 0 disables the auto logon feature.

CAUTION

Automated logon creates a situation in which the computer automatically makes itself available to users without requiring an account name or a password. It is essential, therefore, that this capability be exercised *only* when security is not a concern (if a machine hosts only a single application and is not connected to the network) or if access to the equipment is otherwise controlled.

Automatic Account Lockout

Automatic account lockout disables a user account if a predetermined number of failed logon attempts occurs within a specified time limit. This feature is intended to prevent intrusion by unauthorized users attempting to gain access by guessing a password or launching a dictionary attack. The default setting of Windows 2000 is to allow an unlimited number of failed access attempts to a user account without locking out that account. However, this is not recommended when there is even a remote chance that unauthorized people can gain physical access to logon consoles. (The account lockout feature of Windows 2000 is discussed in Chapter 5.)

Domain Security Concepts and Systems

A **domain** is a collection of computers with centrally managed security and activities. A domain offers increased security, centralized control, and broader access to resources than any other computer system configuration. Security policies are domain-wide controls that specify password requirements, account lockout settings, auditing, user rights, security options, and more.

Domain Security Overview

Domain security is the control of user accounts, group memberships, and resource access for all members of a network instead of for only a single computer. The mechanisms used by Windows 2000 Professional are similar to those employed by domain controllers (that is, Windows 2000 Servers) to manage an entire network's security. All of the information about user accounts, group memberships, group policies, and access controls for resources is contained in Active Directory, a database maintained by one or more domain controllers. A **domain controller** is a Windows 2000 Server system with Active Directory services installed and configured.

Kerberos and Authentication Services

Authentication takes place in a Windows 2000 domain network under two conditions: interactive logon and network authentication. Interactive logon occurs when you press the attention sequence, and then enter your username and password. If you log on to a local system, such as a standalone Windows 2000 Professional system, all authentication is performed by the local security subsystem. If you are logging on to a domain, the local security subsystem communicates with a domain controller using **Kerberos** v5—an authentication encryption protocol—to protect your logon credentials.

A **network authentication** occurs when you attempt to connect to or access resources from some other member of the domain network. Network authentication is used to prove that you are a valid member of the domain, your user account is properly authenticated, and that you have access permissions to perform the requested action. The communications that occur during network authentication are protected by one of several methods, including:

- Kerberos v5
- Secure Sockets Layer/Transport Layer Security (SSL/TLS)
- NTLM (NT LAN Manager) authentication for compatibility with Windows NT 4.0

TIP The authentication protection method is determined by either the communications mechanism (such as IIS or a standard network connection) or the settings in the Local Security Policy. Only one is used, but all can be "active" at one time, so as the client requests or uses one or the other, the server can actively respond. The server responds with the same authentication scheme requested by the client.

Kerberos v5 Authentication

Windows 2000 uses Kerberos v5 as the primary protocol for authentication security. The system uses this protocol to verify the identity of both the client (user) and server (network service or application) upon each resource access. This is known as mutual authentication. It protects the server from unauthorized clients and prevents the user from accessing the wrong or spoofed servers (a spoofed server is one that is programmed to appear to be a particular server when it is another; this is most commonly used on the Internet when an attacker is trying to gain access to credit card information).

The Kerberos authentication system was designed to allow two parties to exchange private information across an open network, such as the Internet. Kerberos assigns a unique key, called a ticket, to each user who logs on to the network. This unique ticket is then embedded in messages to identify the sender of the message to the message's recipient. The Kerberos process is completely invisible to the user. For more information on Kerberos, consult the *Windows 2000 Resource Kit*.

Secure Sockets Layer/Transport Layer Security (SSL/TLS)

Secure Sockets Layer/Transport Layer Security (SSL/TLS) is an authentication scheme often used by Web-based applications and supported in Windows 2000 via IIS (Internet Information Services). SSL functions by issuing an identity **certificate** to both the client and server. A third-party certificate authority that both the client and server have chosen to trust, such as VeriSign (*http://www.verisign.com*), issues these certificates. When a resource request is made, the client sends its certificate to the server. The server verifies the validity of the client certificate, then sends its own certificate to the client along with an encryption key. The client verifies the validity of the server certificate, then uses the encryption key to initiate a communication session with the server. This encrypted communication link is used for all future communications during this session. Once the session is terminated, the link must be rebuilt by starting over with the client sending its certificate to the server.

For more information on SSL, please consult the *Windows 2000 Resource Kit* and the *IIS Resource Kit*.

NTLM

NTLM (NT LAN Manager) authentication is the mechanism used by Windows NT 4.0. Windows 2000 supports this authentication method solely for backward compatibility with Windows NT Server and Windows NT Workstation. NTLM functions by using a static encryption level (40-bit or 128-bit) to encrypt traffic between a client and server.

For more information on NTLM, please consult the *Windows 2000 Resource Kit* or the *Windows NT 4.0 Resource Kit*.

LOCAL COMPUTER POLICY

Another security control built into Windows 2000 is the **local computer policy**. This policy is a combination of controls that in Windows NT existed only in the Registry, via system policies, or as Control Panel applets. Sometimes the local computer policy is called a software policy, an environmental policy, or even a Windows 2000 policy. No matter what name is actually used, the local computer policy is simply the group policy (see Chapter 5) of the local system. The effective local computer policy is the result of the combination of all group policies applicable to the system.

In a Windows 2000 domain network environment, the local computer policy is controlled on a domain basis on a Windows 2000 Server domain controller. This control is based on site, domain, and organizational unit group policies. On a Windows 2000 Professional system, you must manually launch the MMC and add the Global Policy snap-in to manage or change the local computer policy. Once the Global Policy snap-in is loaded, it is displayed as Local Computer Policy. You cannot manage domain policies from a Windows 2000 Professional machine.

The contents of the local computer policy are determined during installation and based on system configuration, existing devices, and selected options and components. Custom policies can be created through the use of .adm files, such as those used by the Windows NT 4.0 System Policy Editor. Such files from Windows NT 4.0 can be used with Windows 2000 with some caveats. When you open and edit the local group policy, you are working with the System.adm file. The .adm files used by the Group Policy editor reside in the \Inf subfolder of the main Windows 2000 directory. Third-party software vendors can use custom .adm files to add additional environmental controls based on their software or services. To learn about creating custom .adm files, see the *Windows 2000 Resource Kit*.

The local computer policy is divided into two sections (see Figure 6-1): Computer Configuration and User Configuration. The Computer Configuration section contains controls that focus on the computer, such as hardware and software settings. The User Configuration section contains controls that focus on the user and the user environment, such as permissions and desktop settings.

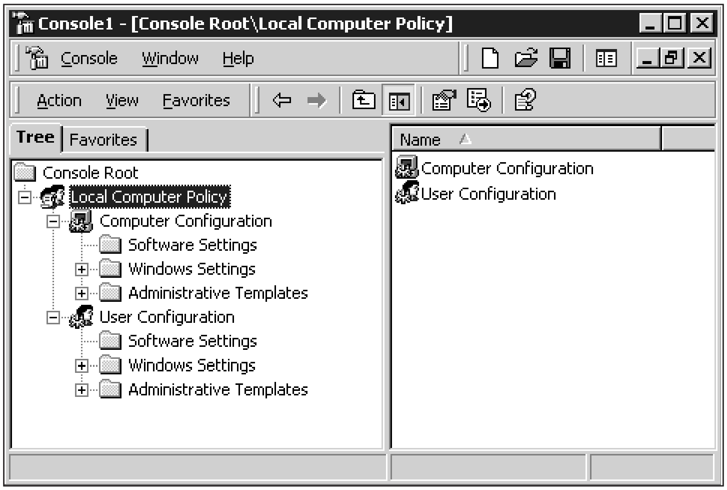


Figure 6-1 The Local Computer Policy snap-in

Because the local computer policy contains over 300 individual controls, you should take the time to peruse the entire collection level by level. (Try Hands-on Project 6-1 to view the local computer policy.)

Computer Configuration

The Computer Configuration section of the local computer policy contains three subfolders: Software Settings, Windows Settings, and Administrative Templates. Software Settings is empty by default; most third-party add-in application settings appear in this folder. The Windows Settings folder contains two items: Scripts and Security Settings. The Scripts item allows you to define one or more scripts to be automatically executed at system startup or shutdown. Security Settings is a container for settings for Account Policies, Local Policies (Audit, User Rights, and Security Options), Public Key Policies, and IP Security Policies. Account Policies and Local Policies were discussed in Chapter 5; Public Key Policies and IP Security Policies are discussed in the following sections. The Administrative Templates folder contains a multilevel collection of computer-related controls (see the "Administrative Templates" section later in this chapter).

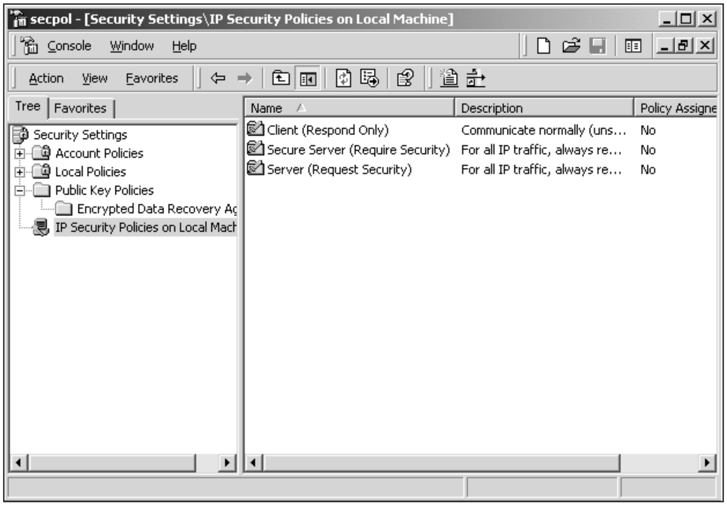
Public Key Policies

There are three purposes for using the **public key policies** controls: to offer additional controls over the encrypting file system (EFS), to enable the issuing of certificates, and to allow you to establish trust in a certificate authority. Please consult the *Windows 2000 Resource Kit* for complete details. (Try Hands-on Project 6-5 to encrypt files with EFS.)

IP Security Policies

IP Security (IPSec) is a security measure added to TCP/IP to protect communications between two systems using that protocol. Windows 2000 is one of the first operating systems to include native support for IPSec. IPSec negotiates a secure encrypted communications link between a client and server through the management of public and private encryption keys. IPSec policies govern how a system communicates via TCP/IP based on your defined security needs. Windows 2000 includes three predefined IPSec policies; however, you can create and manage your own custom IPSec policies. None of the predefined IPSec policies are enabled or assigned by default. For information on creating custom IPSec policies, consult the *Windows 2000 Resource Kit*.

The three predefined IPSec policies are Client (Respond Only), Server (Request Security) and Secure Server (Require Security). See Figure 6-2. The Client (Respond Only) policy is for systems that do not require secure communications at all times. This policy initiates a secure communications link only when another system requests it. This policy does not initiate secure communications by default. The Server (Request Security) policy is for systems that need to use secure communications most of the time. This policy always requests that communications be secured, but allows unsecured communications to occur if IPSec is not available on the other system. The Secure Server (Require Security) policy is for systems that require secure communications at all times. This policy allows communications only if the remote system offers IPSec. Each of these policies can be modified via its Properties dialog box. However, Microsoft recommends creating new policies instead of modifying the default policies.

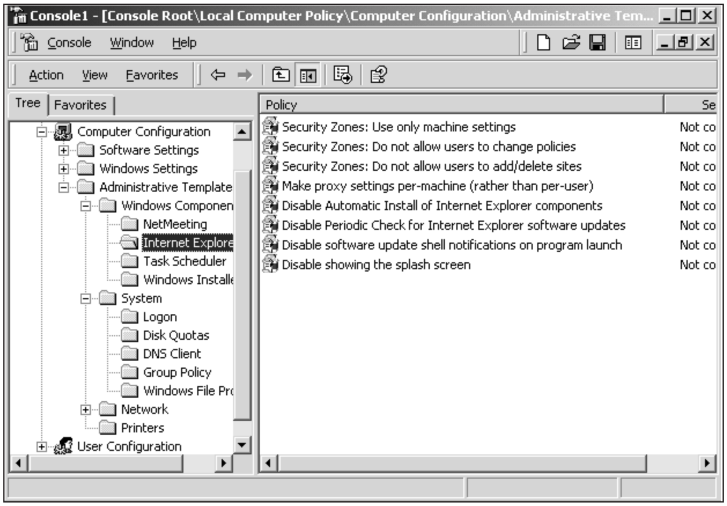

secpol - [Security Settings\IP Security Policies on Local Machine]
Console
Window
Help
Action
View
Favorites
Tree
Favorites
Security Settings
Account Policies
Local Policies
Public Key Policies
Encrypted Data Recovery Ag
IP Security Policies on Local Mach

Name	Description	Policy Assigne
Client (Respond Only)	Communicate normally (uns...	No
Secure Server (Require Security)	For all IP traffic, always re...	No
Server (Request Security)	For all IP traffic, always re...	No

Figure 6-2 Setting IP security policies

Administrative Templates

Administrative templates offer controls on a wide range of environmental functions and features. An administrative template is simply a collection of predefined security and operational controls for Windows 2000. As shown in Figure 6-3, the Administrative Templates folder in the Computer Configuration node of the Local Computer Policy snap-in contains folders and subfolders with specific control items focused on a single aspect of the computer or environmental function.



Policy	Se
Security Zones: Use only machine settings	Not co
Security Zones: Do not allow users to change policies	Not co
Security Zones: Do not allow users to add/delete sites	Not co
Make proxy settings per-machine (rather than per-user)	Not co
Disable Automatic Install of Internet Explorer components	Not co
Disable Periodic Check for Internet Explorer software updates	Not co
Disable software update shell notifications on program launch	Not co
Disable showing the splash screen	Not co

Figure 6-3 The Administrative Templates folder

The controls available through the Administrative Templates folder under computer configurations include:

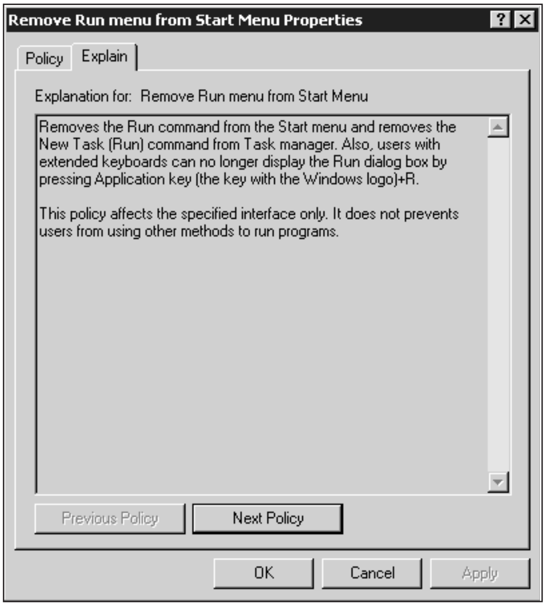
- Controlling security and software updates for Internet Explorer
- Controlling access and use of the Task Scheduler and Windows Installer
- Controlling logon security features and operations
- Controlling disk quotas
- Managing the way group policies are processed
- Managing system file protection
- Managing offline access of network resources
- Controlling printer use and function

User Configuration

The User Configuration portion of the local computer policy is structured in much the same way as the Computer Configuration portion. The User Configuration folder is also divided into three subfolders: Software Settings, Windows Settings, and Administrative Templates. Software Settings is empty by default. Any user-specific Microsoft or third-party software settings appear in this folder. The Windows Settings folder contains three items: Internet Explorer (IE), Scripts, and Security Settings. The Internet Explorer section is used to control user-specific activities of IE, such as the browser interface appearance, connection methods, links, and security zones. The Scripts item allows you to define one or more scripts to automatically execute at user logon or logoff. Security Settings is initially empty, but can become a container for user-specific (rather than computer-specific) security controls. The Administrative Templates folder contains a multilevel collection of user-specific functional and environmental controls. Keep in mind that these controls are for the local computer only. If the computer is a member of a domain, some of these controls may be overwritten by a group, computer, or organizational unit policy from the domain. The items contained in the User Configuration's Administrative Templates section include:

- Internet Explorer configuration, interface, features, and functions controls
- Management of Windows Explorer (interface, available commands, features)
- Management of the MMC
- Control over the Task Scheduler and Windows Installer
- Control over Start menu and taskbar features
- Management of the desktop environment
- Management of the Control Panel applets
- Control of offline network access
- Management of network connections
- Control over logon and logoff scripts and application of group policy

TIP For more information on any control in the Local Computer Policy, open its Properties dialog box and view the Explain tab (see Figure 6-4).


Remove Run menu from Start Menu Properties

Explanation for: Remove Run menu from Start Menu

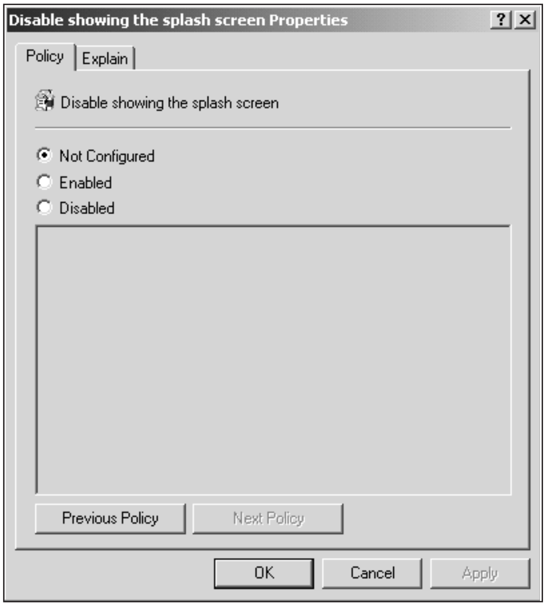
Removes the Run command from the Start menu and removes the New Task (Run) command from Task manager. Also, users with extended keyboards can no longer display the Run dialog box by pressing Application key (the key with the Windows logo)+R.

This policy affects the specified interface only. It does not prevents users from using other methods to run programs.

Figure 6-4 The Explain tab of a Local Computer Policy control dialog box

The Policy tab on the Properties dialog box for each control (see Figure 6-5) offers three settings:

- *Not Configured*—The default for all controls; does not change the existing setting of this control
- *Enabled*—Enables the function or restriction of this control
- *Disabled*—Disables the function or restriction of this control



Disable showing the splash screen Properties

Disable showing the splash screen

● Not Configured
○ Enabled
○ Disabled

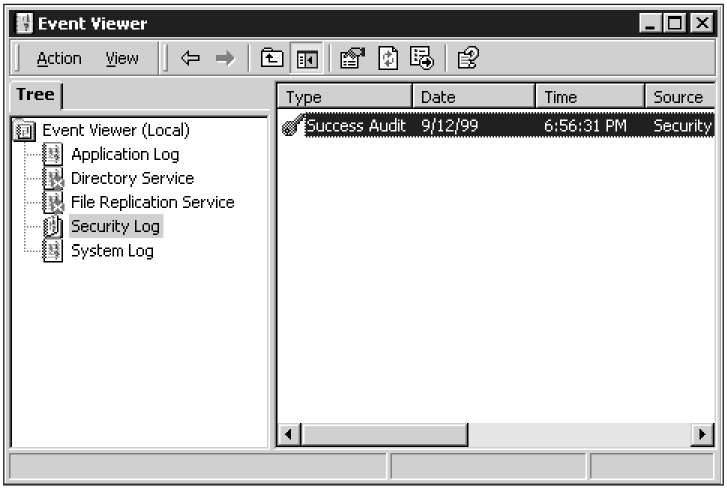
Figure 6-5 The Policy tab on the Properties dialog box

By carefully reading the materials on the Explain tab, you'll be able to understand which of the three settings for each control makes the most sense for the action you wish to enforce or allow. In some cases, selecting the Enable control reveals additional controls, such as selection lists, numerical entry fields, or text entry fields, which provide the additional settings required by some controls. For example, the timeout settings require you not only to enable the control but also to define a time period in minutes or seconds for that timeout period.

AUDITING

Auditing is the security process that records the occurrence of specific operating system **events** in a Security log. Every object in the Windows 2000 system has audit events related to it. These events can be recorded on a success or failure basis and in some cases according to users or groups. For example, logging all failed logon attempts may warn you when an attack attempting to breach your security is occurring; monitoring classified documents for read access can let you know when and who is accessing them. Auditing can provide valuable information about security breaches, resource activity, and user adeptness. Auditing is also useful for investigating performance and planning for expansion.

Auditing is enabled through the Local Security Policy (see Chapter 5). Once enabled, the audited events are recorded in the Security log of the **Event Viewer**. The Event Viewer is accessed via the Administrative Tools (accessed from the Start menu or Control Panel). The Event Viewer maintains logs about application, security, and system events on your computer, enables you to view and manage the logs, to gather information about hardware and software problems, and monitor Windows 2000 security events. To view the items related to auditing, select the Security Log node (see Figure 6-6). Double-clicking an event opens the Event dialog box (see Figure 6-7). This particular audit event records the data about a successful logon of the Administrator account on the workstation named W2KPRO. Audit entries in the Security log contain information about the event, including user logon identification, the computer used, time, date, and the action or event that instigated an audit.

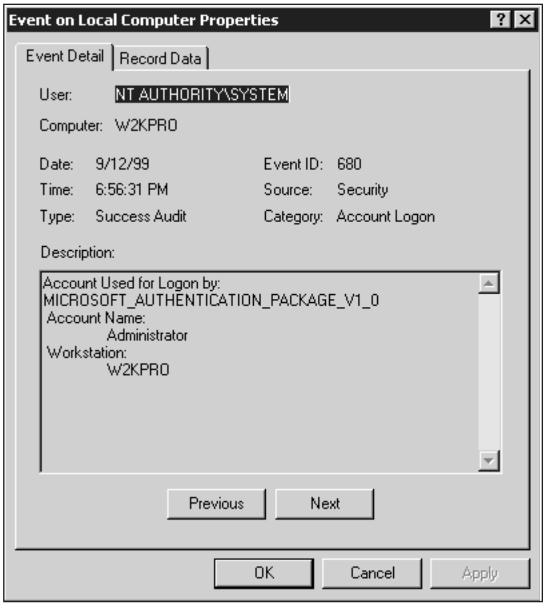


Type	Date	Time	Source
Success Audit	9/12/99	6:56:31 PM	Security

Figure 6-6 An event from the Security Log

If you select the option to audit object access on either a Success or Failure basis, you can define the actions or activities to audit for objects on an object-by-object basis for each possible action, based on that object's type, for specific users and groups. For example, you might audit access to certain network resources, such as files or printers, by different users and/or groups. To set an object's auditing controls:

1. Open the properties dialog box for an NTFS object (such as a file, folder, or printer). Right-click the object, then select **Properties** from the menu.
2. Select the **Security** tab.
3. Click the **Advanced** button.



User: NT AUTHORITY\SYSTEM

Computer: W2KPRO

Date: 9/12/99 Event ID: 680

Time: 6:56:31 PM Source: Security

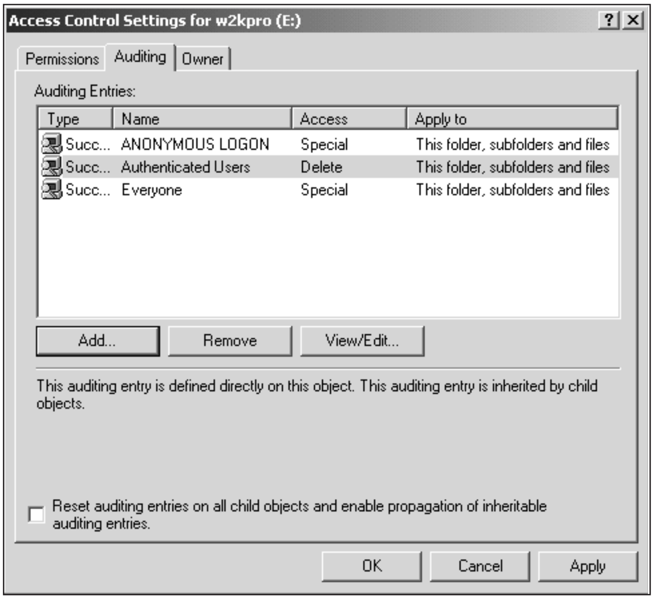
Type: Success Audit Category: Account Logon

Description:

```
Account Used for Logon by:
MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
 Account Name:
          Administrator
 Workstation:
          W2KPRO
```

Figure 6-7 The Event dialog box

4. Select the **Auditing** tab as shown in Figure 6-8. This displays all of the currently defined audit events for this object. It is blank by default.



Auditing Entries:

Type	Name	Access	Apply to
Succ...	ANONYMOUS LOGON	Special	This folder, subfolders and files
Succ...	Authenticated Users	Delete	This folder, subfolders and files
Succ...	Everyone	Special	This folder, subfolders and files

This auditing entry is defined directly on this object. This auditing entry is inherited by child objects.

□ Reset auditing entries on all child objects and enable propagation of inheritable auditing entries.

Figure 6-8 The Auditing tab

5. Click the **Add** button.
6. Select a computer, a group, or a user from the Select User, Computer, or Group dialog box.
7. Click **OK**.
8. Select either Successful or Failed for any of the listed actions for this object type. The selections made here are the actions that are recorded in the Security log.

TIP

If you selected the option to record only Failures in the Local Security Policy, selecting Successful actions in this dialog box does not record items in the Security log.

9. Click **OK**.
10. Repeat Steps 5 through 9 for all users, computers, or groups you wish to audit.
11. Repeat Steps 1 through 10 for all objects.
12. Click **OK** to exit the Access Control Settings dialog box.
13. Click **OK** to exit the Properties dialog box.

TIP

Auditing access by the Everyone group ensures that all access to a user right, object, etc. is audited.

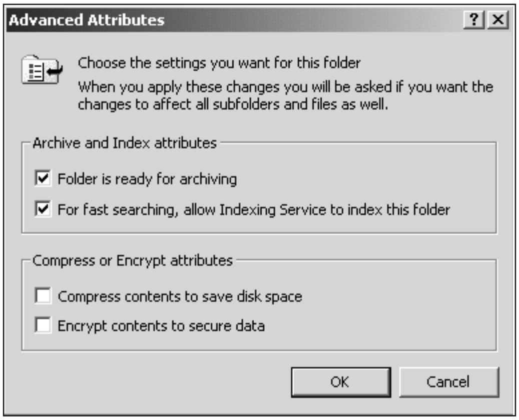
CAUTION

Auditing numerous objects or events can result in a large Security log and can slow down network or computer performance.

The Event Viewer can be configured to monitor the size of the Security log and to take action when it reaches a target size. The actions are Overwrite Oldest Entries as Needed, Overwrite Entries a Specified Number of Days Old, or Never Overwrite. If the maximum size is reached and Never Overwrite is selected, an alert appears stating that the log needs to be cleared. To access these controls, select Properties from the menu that appears when you right-click the log in the Event Viewer.

Encrypting File System (EFS)

Microsoft has extended the native NTFS file system to include encrypted storage. This new security measure, the **encrypting file system (EFS)**, enables you to encrypt data stored on an NTFS drive. When EFS is enabled on a file, folder, or drive, only the enabling user can gain access to the encrypted object. The EFS is enabled through a check box accessed through the Advanced button on the General tab of an object's Properties dialog box (see Figure 6-9).



Advanced Attributes

Choose the settings you want for this folder

When you apply these changes you will be asked if you want the changes to affect all subfolders and files as well.

Archive and Index attributes

☑ Folder is ready for archiving

☑ For fast searching, allow Indexing Service to index this folder

Compress or Encrypt attributes

☐ Compress contents to save disk space

☐ Encrypt contents to secure data

Figure 6-9 Enabling EFS

EFS employs a public and private key encryption method. The private key is assigned to a single user account. No other user (other than Local Administrators or Domain Administrators who are recovery agents by default), computer, or operating system can gain access to the encrypted files. For the authorized user (that is, the user with the correct private key), access to the encrypted files is unhindered. In fact, the entire encryption process is invisible to the user.

TIP

Keep in mind that encryption is just another attribute of NTFS, and you should treat encryption in the same manner as attributes and permissions. Any new file created or copied into an encrypted folder assumes the settings of that folder. Moving an encrypted file to a nonencrypted folder on the same volume allows the file to retain its original settings, but copying the encrypted file causes the file to assume the settings of the destination folder. Because EFS is an additional level of processing required by the operating system to grant access to file-level objects, the performance of the file system can be noticeably impaired. You'll need to perform your own baseline comparison of your storage system's performance to determine exactly how much degradation is caused by EFS.

Windows 2000 includes a command-line tool for batch processing of encryption (that is, encrypting or decrypting large numbers of files or folders via a command-line or batch file). The CIPHER command has the following syntax:

```
CIPHER [/E ,  /D] [/S[:dir]] [/A] [/I] [/F] [/Q] [filename [...]]
```

The following list defines each of the CIPHER command's parameters:

- /E—Encrypts the listed filename(s)
- /D—Decrypts the listed filename(s)
- /S—Performs the action on all subcontents
- /I—Ignores errors and proceeds with processing

- /F—Forces encryption, even on already encrypted files
- /Q—Silences activity except for essential feedback
- *filename*—Specifies a pattern, file, or directory. Wildcards can be used; each pattern must be separated by a space.

When CIPHER is used with only a filename and without parameters, the status of the object is displayed, indicating whether the object is encrypted and whether new files added to a folder will be encrypted.

The primary benefit of EFS is that if your computer is either physically accessed or stolen, the data is protected as long as the malicious user does not gain access to the username and password that holds the private key for the encrypted files. The primary drawback is the increased processing power required to encrypt all writes and decrypt all reads on the fly. This will negatively affect performance to a noticeable extent on many systems.

Chapter Summary

- ❐ Windows 2000 has object-level access controls that provide the foundation for all resource access. By comparing the access control lists associated with individual objects to the access tokens that define the rights of any user process, Windows 2000 decides which object access requests to grant and which to deny.
- ❐ The Windows 2000 logon process (WinLogon) strictly controls how users identify themselves and log on to a Windows 2000 machine. The attention sequence (Ctrl+Alt+Delete) guarantees that an unauthorized user cannot obtain system access. Likewise, WinLogon's protected memory structures prevent this all-important gatekeeper function from being replaced by would-be system crackers. Authentication can take place using various encryption schemes, including Kerberos, SSL, or NTLM.
- ❐ WinLogon also supports a number of logon controls: handling of a default logon name, providing security notices, changing the default shell, handling system shutdown options, and enabling automatic logon. Key local computer policy settings can be used to block unauthorized break-in attempts.
- ❐ The local computer policy controls many aspects of the security system as well as enabling or restricting specific functions and features of the operating system. You can use Windows 2000 auditing capabilities to track down errant behavior or detect when system problems may be occurring. Encrypting file system (EFS) protects your data by an encryption system. All in all, Windows 2000 offers a secure operating environment that is designed to help administrators keep their important assets safe from harm and unwanted exposure.

KEY TERMS

access control list (ACL) — A list of security identifiers that are contained by a resource object. Only those processes with the appropriate access token can activate the services of that object.

access token — Objects containing the security identifier of an active process. These tokens determine the security context of the process.

Active Directory — The database that contains information about a domain's user accounts, group memberships, group policies, and access controls for resources.

auditing — The process of tracking events by recording selected types of events in the Security log.

authentication — The process of validating a user's credentials to allow access to certain resources.

certificate — An electronic identity verification mechanism. Certificates are assigned to a client or server by a certificate authority. When communication begins, each side of the transmission can decide to either trust the other party based on its certificate and continue with the communication or not to trust the other party and terminate communication.

domain — A collection of computers with centrally managed security and activities.

domain controller — A specified computer role of a Windows 2000 Server that authenticates domain logons and maintains the security policies and the account database for a domain.

domain security — The control of user accounts, group memberships, and resource access for all members of a network instead of for only a single computer.

encrypting file system (EFS) — A security feature of NTFS under Windows 2000 that allows files, folders, or entire drives to be encrypted. Once encrypted, only the user account that enabled the encryption has the proper private key to decrypt and access the secured objects.

event — Any significant occurrence in the system or in an application that requires users to be notified or a log entry to be added. Types of events include audits, driver failures, user logons, process launchings, and system shutdowns.

Event Viewer — The utility which maintains logs about application, security, and system events on your computer, and enables you to view and manage the event logs, gather information about hardware and software problems, and monitor Windows 2000 security events.

identification — The process of establishing a valid account identity on a Windows 2000 machine by supplying a correct and working domain name (if necessary) and an account name.

IP Security (IPSec) — An encrypted communication mechanism used by TCP/IP to create protected communication sessions. IPSec is a suite of cryptography-based protection services and security protocols.

Kerberos — An authentication encryption protocol employed by Windows 2000 to protect logon credentials.

local computer policy — A Windows 2000 security control feature used to define and regulate security-related features and functions.

local computer security — The control of user accounts, group memberships, and resource access for a single computer.

network authentication — Part of the act of connecting to or accessing resources from some other member of the domain network. Network authentication is used to prove that you are a valid member of the domain, that your user account is properly authenticated, and that you have access permissions to perform the requested action.

NTLM (NT LAN Manager) authentication — The authentication mechanism used on Windows NT that is retained by Windows 2000 for backward compatibility.

object — Everything within the Windows 2000 operating environment is an object. Objects include files, folders, shares, printers, and processes.

password — A unique string of characters that must be provided before a logon or an access is authorized. Passwords are a security measure used to restrict initial access to Windows 2000 resources.

process — The primary unit of execution in the Windows 2000 operating system environment, a process may contain one or more execution threads, all associated with a named user account, SID, and access token. Processes essentially define the container within which individual applications and commands execute under Windows 2000.

public key policy — A security control of Windows 2000 whereby recovery agents for EFS and domain-wide and trusted certificate authorities are defined and configured. These policies can be enforced on a user-by-user basis.

Secure Sockets Layer/Transport Layer Security (SSL/TLS) — A mechanism used primarily over HTTP communications to create an encrypted session link through the exchange of certificates and public encryption keys.

security ID (SID) — A unique name that identifies a logged-on user to the security system. SIDs can identify one user or a group of users.

shell — The default user process that is launched when a valid account name and password combination is authenticated by the WinLogon process for Windows 2000. The default shell of Windows 2000 is Windows Explorer. The default shell process manages the desktop, Start menu, taskbar, and other interface controls. The shell process defines a logged-on user's run-time environment from the point of authentication forward, and supplies all spawned processes or commands with its access token to define their access permissions, until that account logs out.

user account — This entity contains all of the information that defines a user to the Windows 2000 environment.

user profile — A file that saves a user's preferences and environmental settings.

WinLogon — The process used by Windows 2000 to control user authentication and manage the logon process. WinLogon produces the logon dialog box where username, password, and domain are selected, and it controls automated logon, warning text, the display of the shutdown button, and the display of the last user to log onto the system.

REVIEW QUESTIONS

1. Which of the following determines which users and groups have access to a particular Windows NT object?
   a. Security Access Manager
   b. local computer policy
   c. Event Viewer
   d. group policy
2. All processes in Windows 2000 require an access token. True or False?
3. A SID is a unique number and is never duplicated. True or False?
4. Permissions that are changed while the user is actively logged on do not take effect until that user logs on to the system again. True or False?
5. The default Windows 2000 authentication method requires the user to supply valid domain and account names, plus a valid password; however, Windows 2000 permits use of alternate authentication techniques. True or False?
6. What is the first thing the security system looks for when it scans an ACL for an object?
   a. a Deny to the object for the requested service, at which point access is immediately denied
   b. any ACL that provides the requested permission
   c. It checks the default, and if access is permitted, it allows the request to proceed.
   d. none of the above
7. Windows 2000 ________________ access to new objects by default.
   a. restricts
   b. allows
8. Which of the following is a good reason for adding DontDisplayLastUserName to the Windows 2000 Registry? (Choose all that apply.)
   a. to prevent easy discovery of user account names
   b. to improve security on a shared machine
   c. to reduce burnout on the machine's monitor
   d. to force users to provide a valid username in addition to a password to logon
9. The Windows 2000 authentication process can be automated by adding default user information and the ________________ value to the Registry.
   a. DontDisplayLastUsername
   b. AutoAdminLogon
   c. Legal Notice Caption
   d. AutomateLogon

10. Which of the following is the most likely reason to have a security notice appear when users attempt to log on to a Windows 2000 machine at the National Security Agency?
    a. to make sure that outsiders don't try to break into the system
    b. to inform unauthorized users that they are subject to legal action if they obtain unauthorized access to the system
    c. to remind valid system users about Acceptable Use Policies
    d. none of the above
11. The default shell process for Windows 2000 is called the:
    a. Windows Explorer
    b. Program Manager
    c. command shell
    d. C shell
12. The ________________ is created by the Windows 2000 security subsystem at logon and identifies the current user to the subsystem.
    a. access ID
    b. security ID
    c. group ID
    d. access token
13. The ________________ key sequence initiates the logon process.
    a. Ctrl+Esc
    b. Alt+Tab
    c. Ctrl+Break
    d. Ctrl+Alt+Delete
14. An access token is required to access any Windows 2000 object. True or False?
15. To customize the security structure of your Windows 2000 system, you can change the behavior of the logon process. True or False?
16. What is the primary protocol that Windows 2000 uses for authentication?
    a. NTLM
    b. Secure Sockets Layer
    c. Kerberos
    d. NetBIOS

17. Which of the following statements are true about the local computer policy? (Choose all that apply.)
    a. It is used to control aspects of the Windows 2000 security system.
    b. It is used to assign user accounts to groups.
    c. It can be customized by third-party applications.
    d. It can be superceded by a domain's group policy.
18. What is the special-purpose application invoked by the Windows 2000 attention sequence that serves as the logon process?
    a. WinPopup.exe
    b. WinLogon.exe
    c. Usermgr.exe
    d. Explorer.exe
19. What security feature is added to Windows 2000 specifically to protect TCP/IP communications?
    a. Kerberos
    b. IPSec
    c. strong passwords
    d. EFS
20. What is EFS used to protect?
    a. passwords
    b. data files
    c. group policy
    d. communication sessions
21. If the 2000 Explorer shell is replaced with the Program Manager shell, which of the following side effects will occur? (Choose all that apply.)
    a. no access to the Start menu
    b. no taskbar
    c. no access to the Task Manager
    d. no more DOS command prompt
22. Only the user who encrypted a file via EFS can access that file later. True or False?
23. What predefined IPSec policy should you use to employ encryption only when required by a remote system?
    a. Client (Respond Only)
    b. Server (Request Security)
    c. Secure Server (Require Security)

24. Auditing can be defined for an object for specific users and groups for one or more individual services or actions. True or False?
25. Audit events are recorded in the System Log. True or False?

Hands-on Projects

Project 6-1

To open the local computer policy:

TIP This project shows you where security controls are managed.

1. Open the Run command (**Start**, **Run**).
2. Type **mmc**, then click **OK**. This launches the Microsoft Management Console.
3. Select **Add/Remove Snap-in** from the Console menu.
4. Click the **Add** button.
5. Locate and select **Group Policy**.
6. Click **Add**.
7. On the Select Group Policy object dialog box, notice that Local Computer is listed by default. Click **Finish**.
8. Click **Close** on the Add Standalone Snap-in dialog box.
9. Click **OK** on the Add/Remove Snap-in dialog box.
10. The Local Computer Policy node should now appear in the MMC.

Project 6-2

To disable the display of the last username on the logon screen:

TIP This project requires that you first complete Hands-on Project 6-1.

1. In the Local Computer Policy console, locate the Computer Configuration node. Click its boxed **plus sign** to expand its contents.
2. Locate the Windows Settings node. Click on its boxed **plus sign** to expand its contents.

3. Locate the Security Settings node. Click on its boxed **plus sign** to expand its contents.
4. Locate the Local Policies node. Click on its boxed **plus sign** to expand its contents.
5. Locate and select the **Security Options** node.
6. In the Details pane, locate and select **Do not display last user name in logon screen**.
7. Select the **Action** menu, then click **Security**. The Local Security Policy Setting dialog box for the selected control is displayed.
8. Select the **Enabled** radio button.
9. Click **OK**.

Project 6-3

To display a legal warning message at logon:

> TIP This project requires that you first complete Hands-on Project 6-1.

1. In the Local Computer Policy console, locate and select the subnode of **Computer Configuration**, **Windows Settings**, **Security Settings**, **Local Policies**, **Security Options**.
2. Locate and select **Message title for users attempting to log on**.
3. Select the **Action** menu, then click **Security**.
4. In the field, type **Warning!** Click **OK**.
5. Select **Message text for users attempting to log on**.
6. Select the **Action** menu, then click **Security**.
7. In the field, type a warning message similar to the following: (*Note*: This excellent security warning message is reproduced from *The Windows NT Security Handbook*, by Tom Sheldon, Osborne/McGraw-Hill: Berkeley, 1997.) **Authorized Users Only! The information on this computer and network is the property of (*name organization here*) and is protected by intellectual property law. You must have legitimate access to an assigned account on this computer to access any information. You are permitted only to access information as defined by the system administrators. Your activities may be monitored. Any unauthorized access will be punished to the full extent of the law.**
8. Click **OK**.

Project 6-4

To change the default shell:

TIP Changing the shell will result in a new user interface. The Program Manager does not offer a Start menu, toolbar, Task Manager, and many other interface controls to which you are accustomed from Windows 2000. Employ this Hands-on Project with caution.

1. Open the Run command (**Start**, **Run**).
2. Type **regedit**, then click **OK**.
3. Locate and select the key:
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon.
4. Locate and select the **Shell** value.
5. Select **Modify** from the Edit menu.
6. Change the value data from Explorer.exe to **Progman.exe**.
7. Click **OK**.
8. Select **Exit** from the Registry menu.The system is now configured to launch the Windows NT 3.51 Program Manager as the shell.
9. To return the shell to Windows Explorer, either reopen the Registry editor now and change the Shell value back to its original setting (Explorer.exe), or, if you have already logged in with Program Manager as the shell, use the Run command from the File menu of the Program Manager to launch Regedit and make the change.

Project 6-5

To encrypt a folder with EFS:

TIP The folder must be on an NTFS file system to complete this exercise.

1. Launch Windows Explorer (**Start**, **Programs**, **Accessories**, **Windows Explorer**).
2. Select the **C** drive in the left column. (If drive C is not formatted with NTFS, select some other drive which is formatted with NTFS.)
3. Select the **File** menu, then **New**, **Folder**.
4. Type a name for the folder (such as **EFStemp**), then press **Enter**.
5. Right-click the new folder, then select **Properties** from the menu.
6. On the General tab, click the **Advanced** button.
7. Click to place a check in the **Encrypt contents to secure data** check box.

8. Click **OK**.
9. Click **OK**.
10. Log off (**Start**, **Shutdown**; select **log off**).
11. Log on to the system with a different user account (Ctrl+Alt+Delete, then provide a different user name and password).
12. Launch Windows Explorer (**Start**, **Programs**, **Accessories**, **Windows Explorer**).
13. Locate and try to access the **EFStemp** folder. Notice that you are unable to gain access.
14. Log off (**Start**, **Shutdown**; select **log off**).
15. Log on with the user account used to encrypt the folder (Ctrl+Alt+Delete, then provide user name and password).
16. Locate and try to access the EFStemp folder. Notice that you are able to gain access.
17. Right-click the EFStemp folder, then select **Properties** from the menu.
18. On the General tab, click the **Advanced** button.
19. Deselect (uncheck) the **Encrypt contents to secure data** check box. Click **OK**.
20. Click **OK**.
21. Click **OK**.

Project 6-6

To explore the local computer policy:

TIP This project requires that you first complete Hands-on Project 6-1.

1. Expand the **Computer Configuration** node of the Local Computer Policy snap-in.
2. Expand the **Administrative Templates** node.
3. Expand each of the **Windows Components**, **System**, **Network**, and **Printers** subnodes.
4. Select each subnode one by one. Review the control details contained in each.
5. To open the Properties of a control detail, select it, select the **Action** menu, and then click **Properties**.
6. View the Policy and Explain tabs of all control details that interest you.
7. Expand the **User Configuration** node and all of its subnodes.
8. Perform the same expansion and exploration as you did under the Computer Configuration node.
9. Select the **Exit** command from the Console menu of the MMC to close the utility. Click **Cancel** to discard any changes, if prompted.

Project 6-7

To set permissions on a file or folder:

TIP This hands-on project requires that Windows 2000 be installed and an NTFS partition is present.

1. Launch Windows Explorer (**Start**, **Programs**, **Accessories**, **Windows Explorer**).
2. In the left pane, select a drive formatted with NTFS within My Computer.
3. In the right pane, select a file or folder.
4. From the **File** menu, select **Properties**.
5. Select the **Security** tab.
6. Click the **Add** button.
7. Select the **Authenticated Users** group.
8. Click **Add**.
9. Click **OK**.
10. Click the **Authenticated Users** group which now appears in the list of names on the Security tab for the NTFS object.
11. Select the **Modify** checkbox in the **Allow** column.
12. Select the **Everyone** group. Notice how the defined permissions for these two groups differ.
13. Click **OK**.

Project 6-8

To enable file access auditing:

1. Open the Control Panel by selecting **Start**, **Settings**, **Control Panel**.
2. Double-click the **Administrative Tools** icon.
3. Double-click the **Local Security Policy** icon.
4. Expand the **Local Policies** node by double-clicking it.
5. Select the **Audit** policy.
6. Double-click the **Audit object access** item.
7. Select the **Success** checkbox. Click **OK**.
8. Launch **Windows Explorer**.
9. Locate and select any text document on your computer, such as **%systemroot%\Winnt\setuplog.txt**.
10. Select the **Properties** command from the **File** menu.
11. Select the **Security** tab.

12. Click **Advanced**.
13. Select the **Auditing** tab.
14. Click **Add**.
15. Select **Authenticated Users**.
16. Click **OK**.
17. Select the **List Folder/Read Data** check box under Successful.
18. Click **OK**.
19. Click **OK**.
20. Click **OK**.
21. Double-click the text file to open it.
22. Close Notepad.
23. Return to Administrative Tools by clicking its button on the taskbar.
24. Double-click the **Event Viewer** icon.
25. Select the **Security** log.
26. Double-click one of the event details.
27. Using the arrow buttons, scroll through the most recent event details to locate an event dealing with the successful reading of the text file.
28. Click **OK** to close the Event detail.
29. Close the Event Viewer.
30. Close Administrative Tools.
31. Close Windows Explorer.
32. On the Local Security Settings dialog box, double-click **Audit object access**.
33. Deselect **Success**.
34. Click **OK**.
35. Close Local Security Settings.

CASE PROJECTS

1. You've been assigned the task of defining a security policy for your company. You've been given basic guidelines to follow. These include preventing users from installing software, securing the logon process, and enforcing disk quotas. Using the Local Computer Policy snap-in, detail the controls you should configure and what settings you think would work best to accomplish these goals.
2. You've recently inherited the responsibility of administering a Windows 2000 network. The last administrator was rather lax in restricting user access. After working through the data folders to correct the access permissions, you suspect that some users still have access to confidential files. What can you do to determine if this type of access is still occurring? Describe the steps involved in enabling this mechanism and examining the results.